



Opdracht 1 - ITSOF3

Groep 5

Kyllian Lissens

Jens Galle

Gianni Selleslagh

| | |
|-----------------------|----------|
| Korte inhoud | 2 |
| Samenvatting | 3 |
| Situatie to-be | 4 |
| Architectuur | 5 |
| Verantwoording | 6 |
| Schema's | 7 |

Korte inhoud

In dit document gaan we bespreken hoe we de analyse en architectuur hebben aangepakt van onze ICT Architecture opdracht.

Samenvatting

Ons doel is aan onze gebruikers een platform aan te bieden waarbij ze op een veilige manier bestanden kunnen delen met andere gebruikers.

In onze applicatie gaan we een API maken dat de webapplicatie gaat gebruiken om de nodige data op te slaan in een database, hiervoor gaan we de infrastructuur van AWS gebruiken.

Om de kosten van de infrastructuur te verminderen gaan bestanden maximum 1 dag bijgehouden worden.

Situatie to-be

Iedereen moet ons platform kunnen gebruiken om tijdelijk een bestand te kunnen uploaden en/of downloaden.

De bestanden zullen beveiligd worden, waardoor niemand de kans krijgt om zomaar iemand anders zijn/haar bestanden te bekijken.

Er zal ook een access log zijn die bijhoudt wie welke file download of opvraagt.

Files zullen vervallen na 24u om hoge kosten tegen te gaan.

We zullen gebruik maken van het AWS platform, zodat we altijd beschikbaar zijn, kunnen bij schalen indien nodig en efficiënt kunnen omgaan met onze resources (geld en dev-time)

Architectuur

Om de bestanden op te slaan gebruiken we een S3 bucket van amazon, met als webserver een EC2 instance.

Door middel van security groups kunnen we de S3 bucket en de EC2 instance aan elkaar koppelen, waardoor we makkelijk met onze endpoints aan onze data te raken.

Dankzij de S3 bucket hebben we native access logs, doordat we een simpele optie aanzetten die de logs naar een aparte bucket stuurt en daar bewaart voor ons.

We zetten ook een Amazon RDS op, om een lijst van bestanden, bestandsnamen, UUIDs bij te houden. We kiezen voor een SQL-oplossing, nl. MariaDB.

Via S3 policies zorgen we ervoor dat alle bestanden een maximum levensduur hebben van 24 uur.

Verantwoording

EC2: We krijgen hier geen keuze in, dit is een voorwaarde van de klant. We kunnen wel EC2 gebruiken om ook onze frontend-client mee te hosten op een EC2 instantie.

S3: We kiezen voor S3 omdat er handige features zijn voor onze use case, waaronder de security access logging feature.

We gebruiken een eigen Authentication API om gebruikers te laten registreren en in te loggen. De data hiervoor slaan we ook op in de RDS. Deze API is geschreven in NodeJS in combinatie met express.

RDS MariaDB: We hebben gekozen voor MariaDB omdat het een open source, gratis relationele database is die onze noden perfect kan vervullen.



